

Amendment and Response

Applicant: Francisco Corella

Serial No.: 09/483,185

Filed: January 14, 2000

Docket No.: 10991054-1

Title: AUTHORIZATION INFRASTRUCTURE BASED ON PUBLIC KEY CRYPTOGRAPHY



IN THE SPECIFICATION

Please replace the paragraph beginning at page 4, line 11 with the following rewritten paragraph, which has all active hyperlinks disabled:

a¹

The Transport Layer Security (TLS) working group of the Internet Society and the Internet Engineering Task Force proposes using attribute certificates. See S. Farrell, *TLS Extensions for Attribute Certificate-Based Authorization*, Internet Draft, August 20, 1998; and Web Page of the TLS Working Group, <http://www.ietf.org/html.charters/tls-charter.html>. An attribute certificate binds a name to authorization information and does not contain a public key. A TLS client would be allowed to present an attribute certificate in addition to an ordinary public key certificate during the initial hand-shake. However, the TLS proposal only applies to the TLS protocol and does not explain how attribute certificates are issued. Thus far, the efforts of the Internet Society and the Internet Engineering Task Force have not yet provided a concrete blue print for solving the authorization problem using public key cryptography.

Please replace the paragraph beginning at page 4, line 24 with the following rewritten paragraph:

a²

The security architecture of Microsoft's ~~Windows 2000~~ Windows 2000® operating system addresses authentication and authorization at the scale of an enterprise network. However, the ~~Windows 2000~~ Windows 2000® operating system security architecture is based on the symmetric-key Kerberos protocol, with public-key enhancements that accommodate smart card authentication. Consequently, the ~~Windows 2000~~ Windows 2000® operating system security architecture is inherently harder to manage, less scalable, and more vulnerable to attack than could be possible if the security architecture was entirely based on public-key cryptography. Moreover, the ~~Windows 2000~~ Windows 2000® operating system security architecture's inter-operability with other Kerberos implementations is limited.

Amendment and Response

Applicant: Francisco Corella

Serial No.: 09/483,185

Filed: January 14, 2000

Docket No.: 10991054-1

Title: AUTHORIZATION INFRASTRUCTURE BASED ON PUBLIC KEY CRYPTOGRAPHY

Please replace the paragraphs beginning at page 5, line 11 with the following rewritten paragraphs:

a³ The security architecture of Microsoft ~~Windows 2000~~ Windows 2000® operating system is based on an extension of Kerberos called Public Key Cryptography for Initial Authentication in Kerberos (PKINIT). PKINIT allows public key cryptography to be used for user authentication. PKINIT makes it possible to use smart cards for authentication. Nevertheless, the Windows® operating system Domain Controller, which serves the role of the Kerberos Key Distribution Center in the ~~Windows 2000~~ Windows 2000® operating system security architecture, must still share symmetric keys with other computers on the network. When these computers are Windows® operating system based machines, the shared symmetric keys are distributed automatically. As in the case of the Kerberos infrastructure itself, this adds complexity and vulnerability to the protocol. When these computers are non-Windows® operating system based machines, the shared symmetric keys have to be installed by hand by the user administrator, which is quite costly and limits scalability.

The ~~Windows 2000~~ Windows 2000® operating system security architecture does not scale well beyond a single domain. Access to machines in other domains relies on trust relationships along a hierarchy or a web of domain controllers, which is difficult to administer and introduces delays.

Kerberos tickets used in the ~~Windows 2000~~ Windows 2000® operating system security architecture carry proprietary security identifiers (SIDs), which are used to obtain access to objects or properties of objects. Since, non-Windows® operating system based machines and applications do not understand SIDs, inter-operability in a heterogeneous environment is limited.
